**PROJECT SPECIFICATION**

**Give Your Application Auto-Deploy Superpowers**

**Submitted By Kapil Chaudhary**

# Section 1: Selling CI/CD to your Team/Organization

**CRITERIA: Explain the fundamentals and benefits of CI/CD to achieve, build, and deploy automation for cloud-based software products.**

Presentation has been prepared and detail available in Presentation.pdf file.

# Section 2: Deploying Working, Trustworthy Software

**CRITERIA: Utilize Deployment Strategies to design and build CI/CD pipelines that support Continuous Delivery processes.**

**A public git repository with your project code [URL01]**

**[URL01] -** [**https://github.com/kapilhubgit/cdond-c3-projectstarter.git**](https://github.com/kapilhubgit/cdond-c3-projectstarter.git)

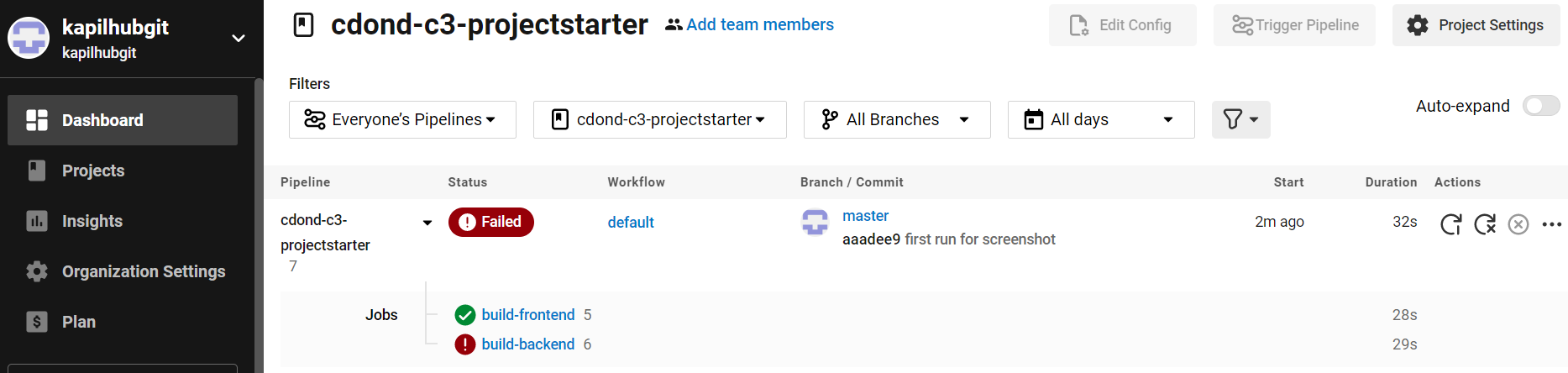
**[URL02] -** [**http://udapeople-139db3a.s3-website-us-east-1.amazonaws.com/#/employees**](http://udapeople-139db3a.s3-website-us-east-1.amazonaws.com/#/employees)

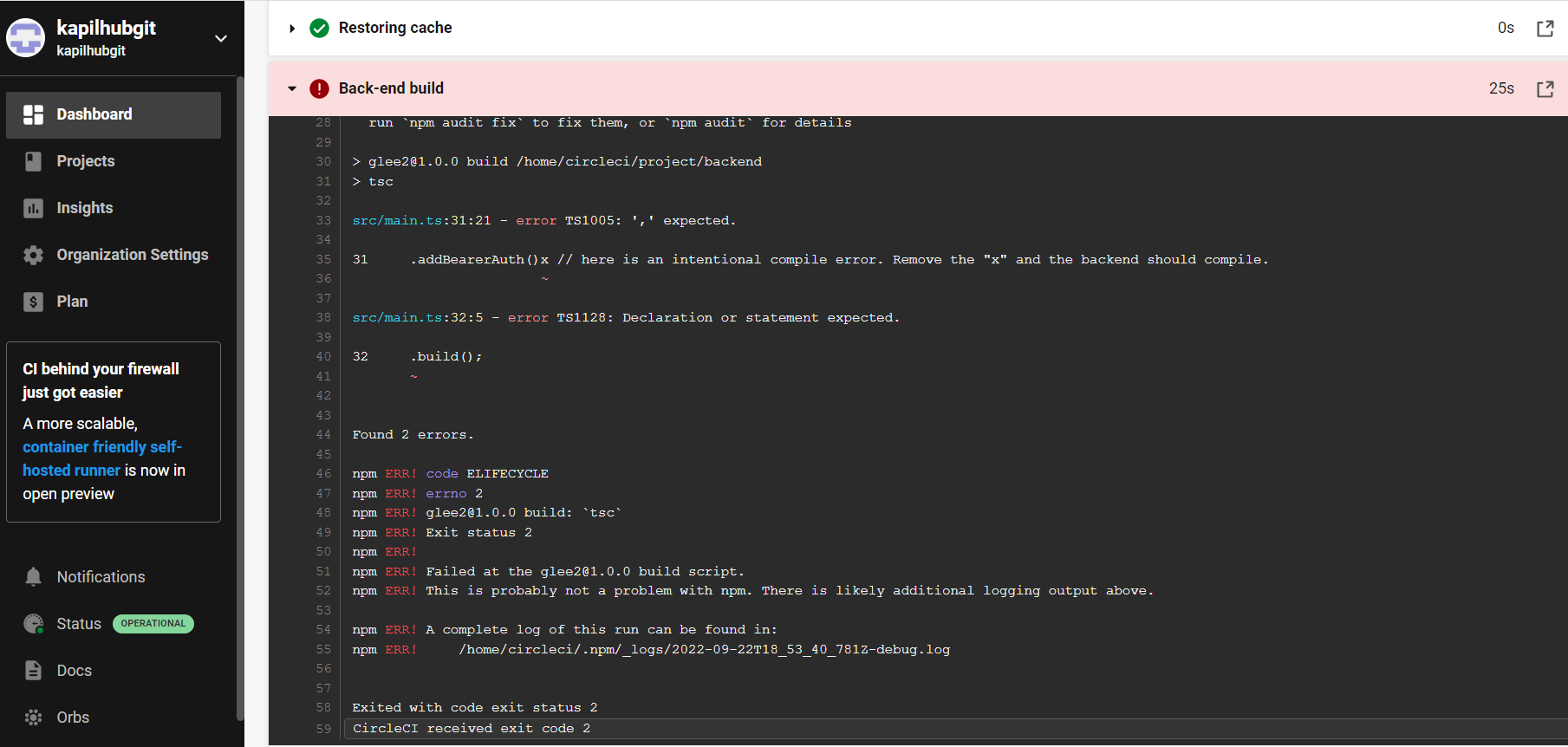
**[URL03] -** [**http://d3283udy6tfw1e.cloudfront.net/#/employees**](http://d3283udy6tfw1e.cloudfront.net/#/employees)

**[URL04] -** [**http://34.205.239.249:3030/api/status**](http://34.205.239.249:3030/api/status)

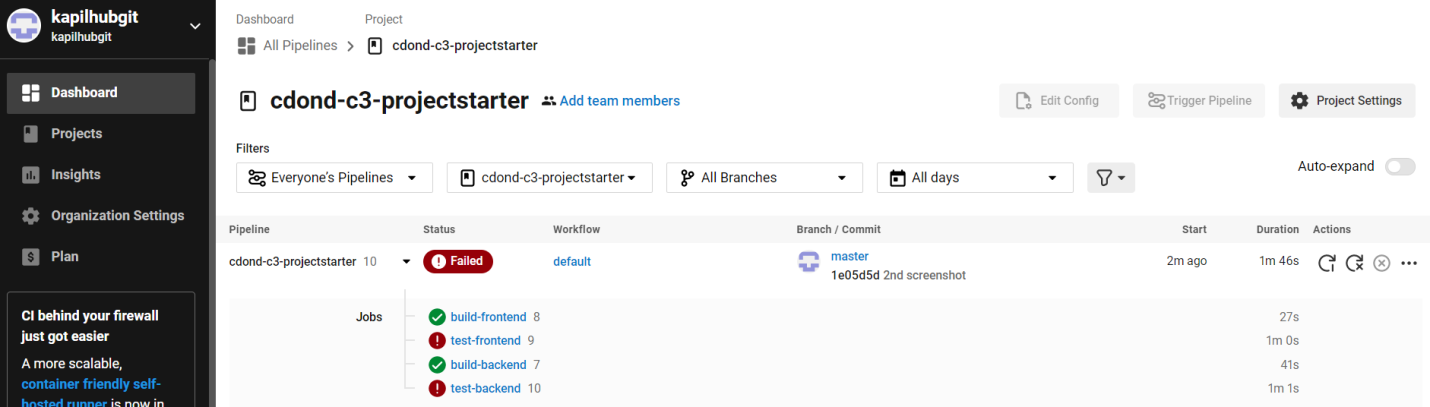
**[URL05] -** [**http://ec2-100-26-190-7.compute-1.amazonaws.com:9090/alerts**](http://ec2-100-26-190-7.compute-1.amazonaws.com:9090/alerts)

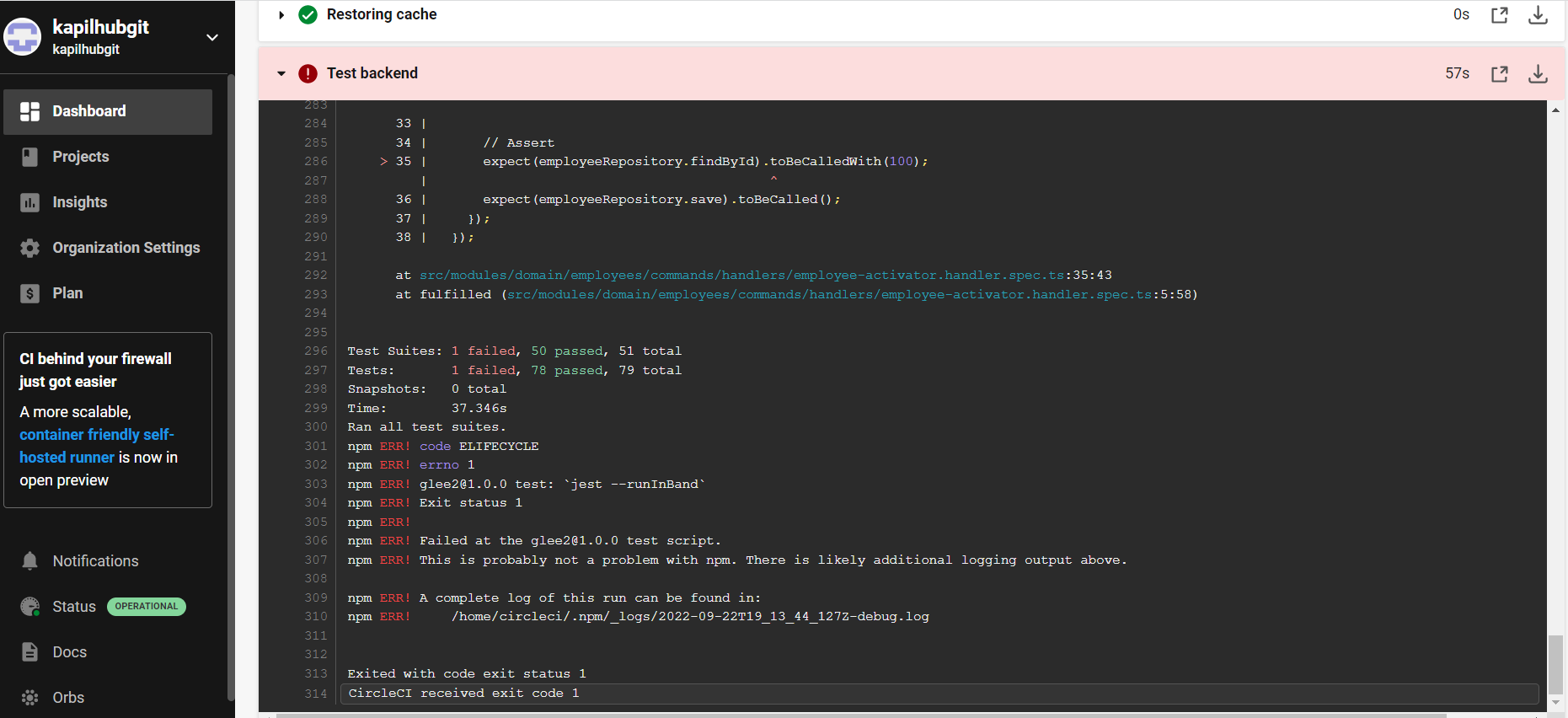
**Build Jobs that failed because of compile errors. [SCREENSHOT01]**

****

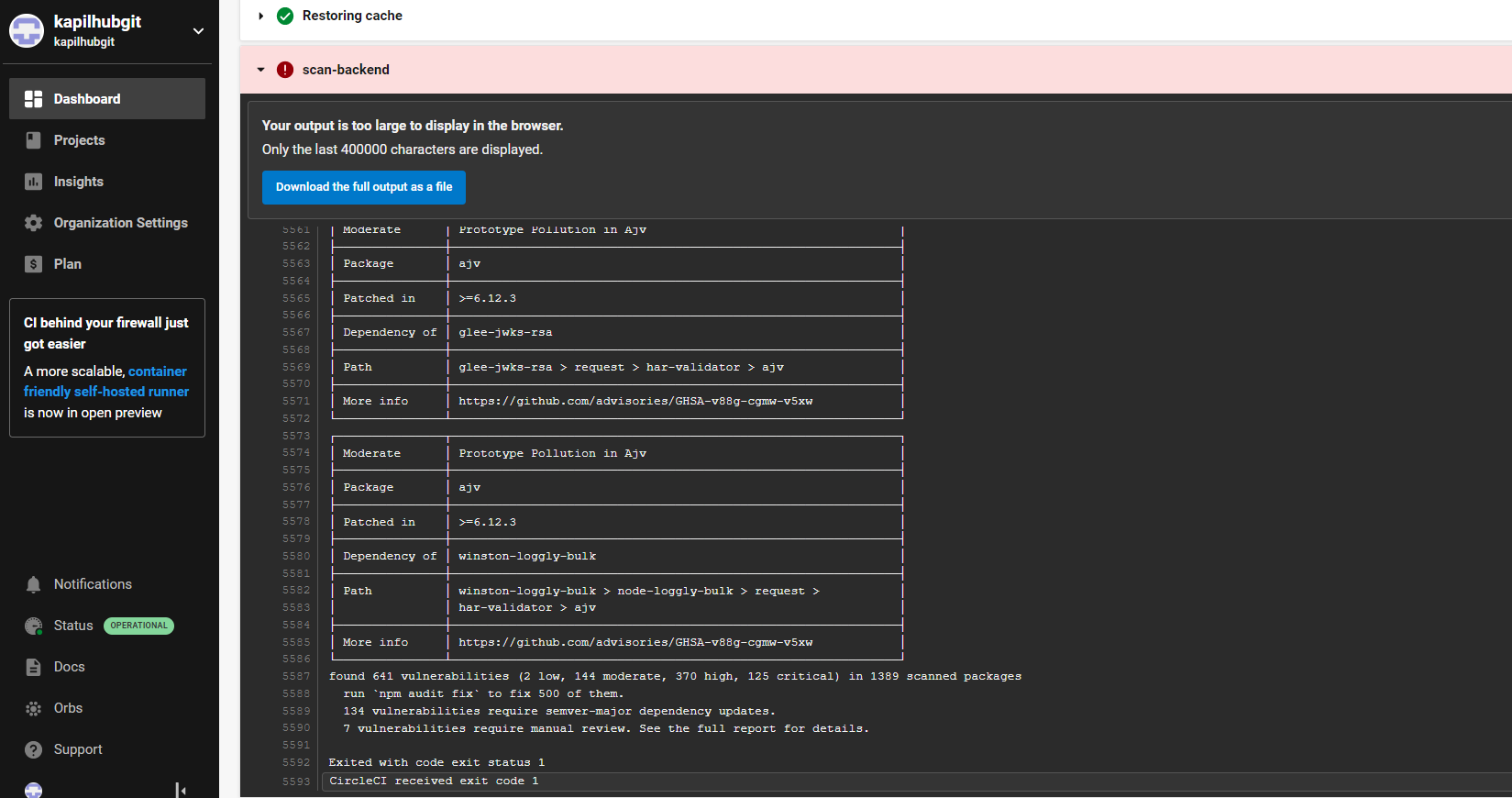
****

**Failed unit tests. [SCREENSHOT02]**

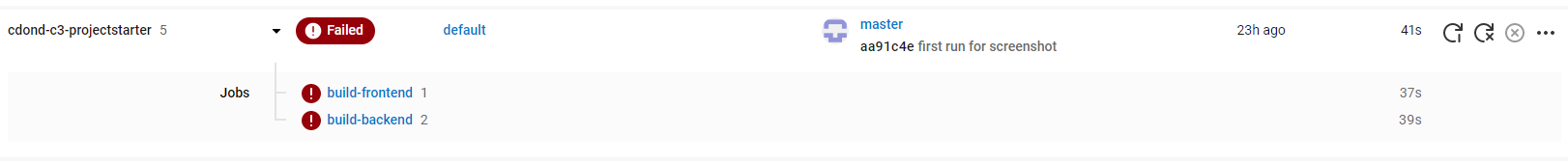
****

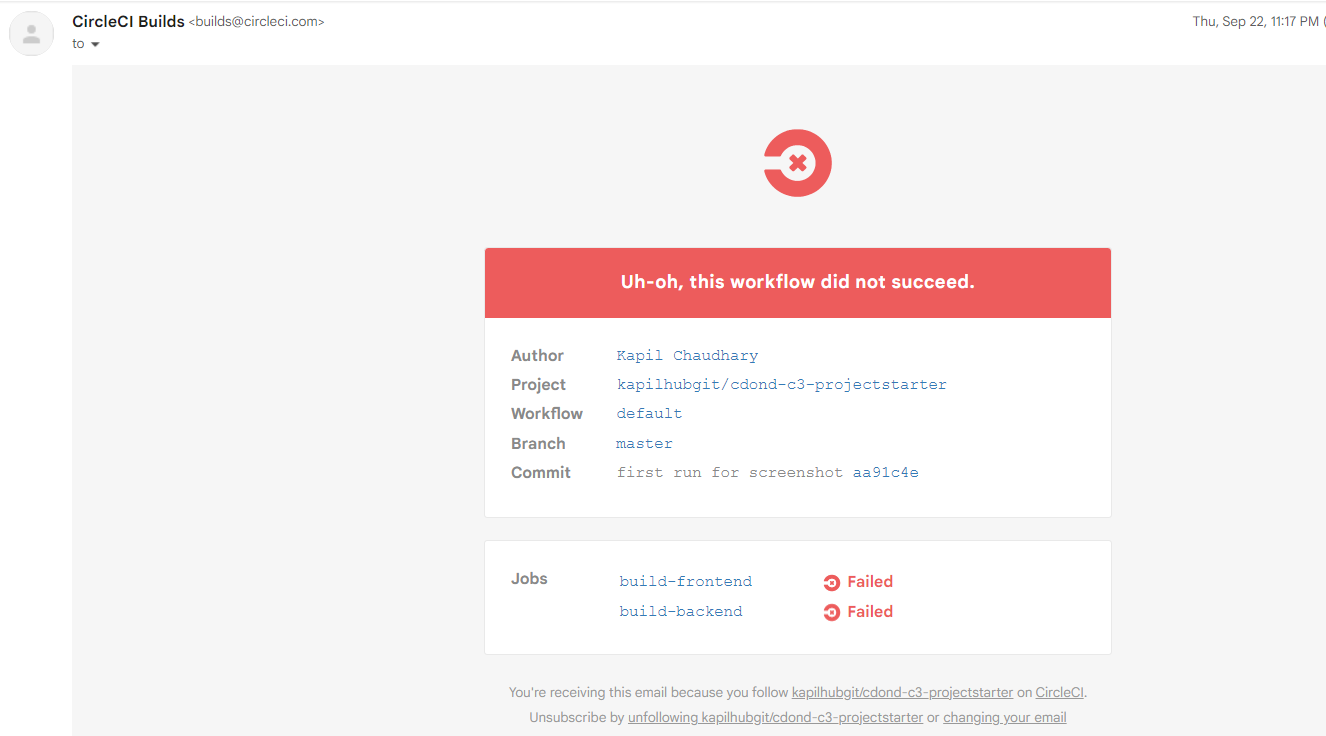
****

**Failure because of vulnerable packages. [SCREENSHOT03]**

****

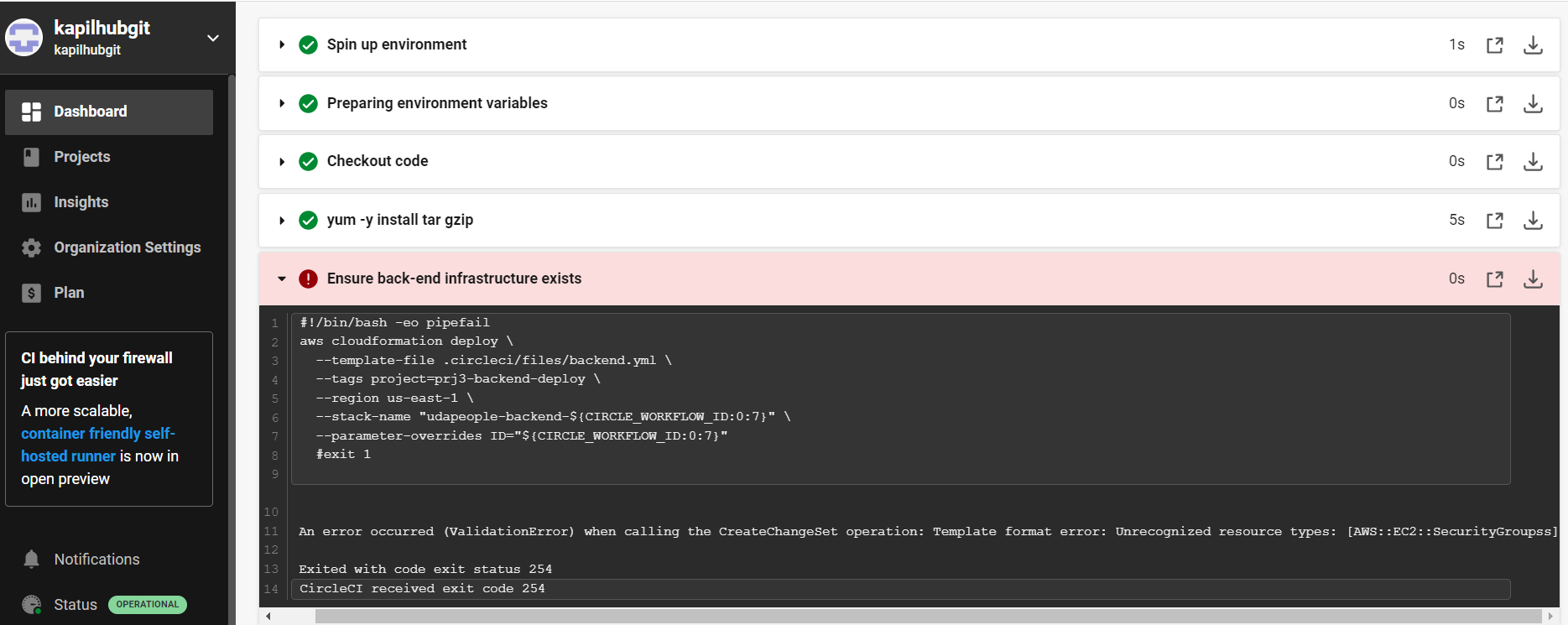
**An alert from one of your failed builds. [SCREENSHOT04]**

****

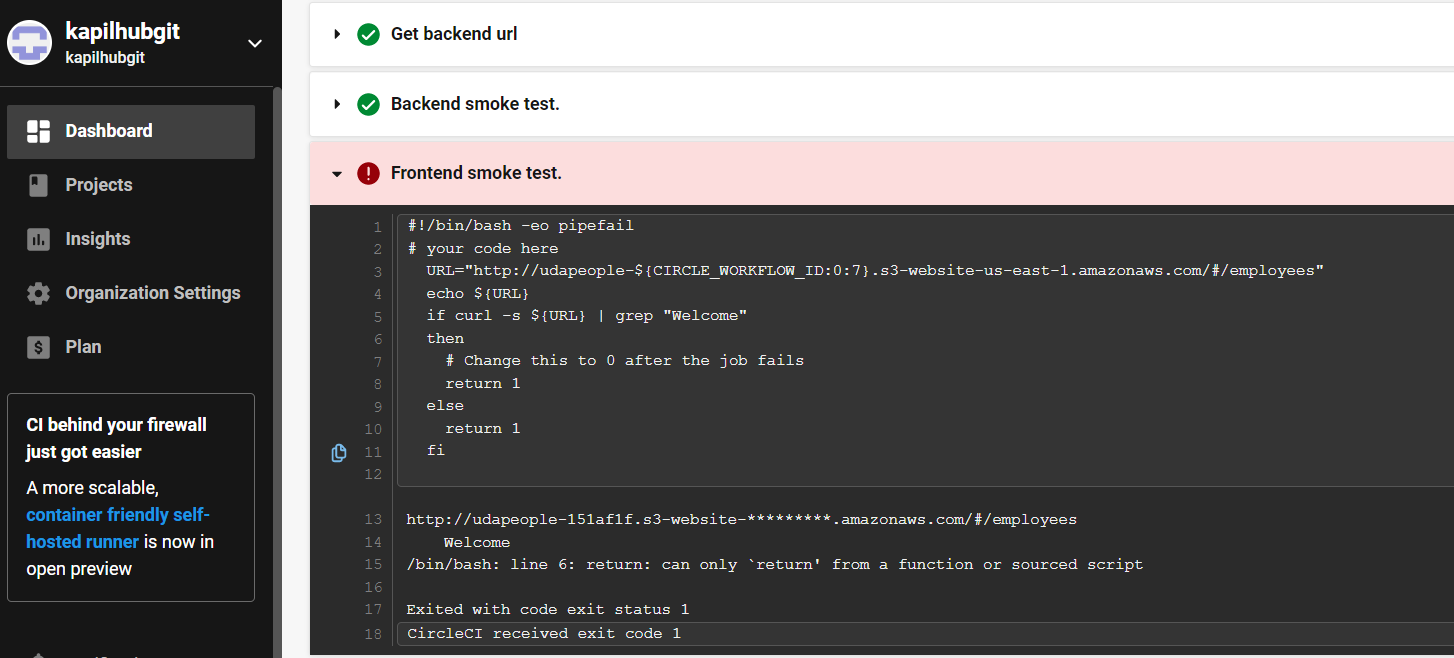
****

**CRITERIA: Utilize a configuration management tool to accomplish deployment to cloud-based servers.**

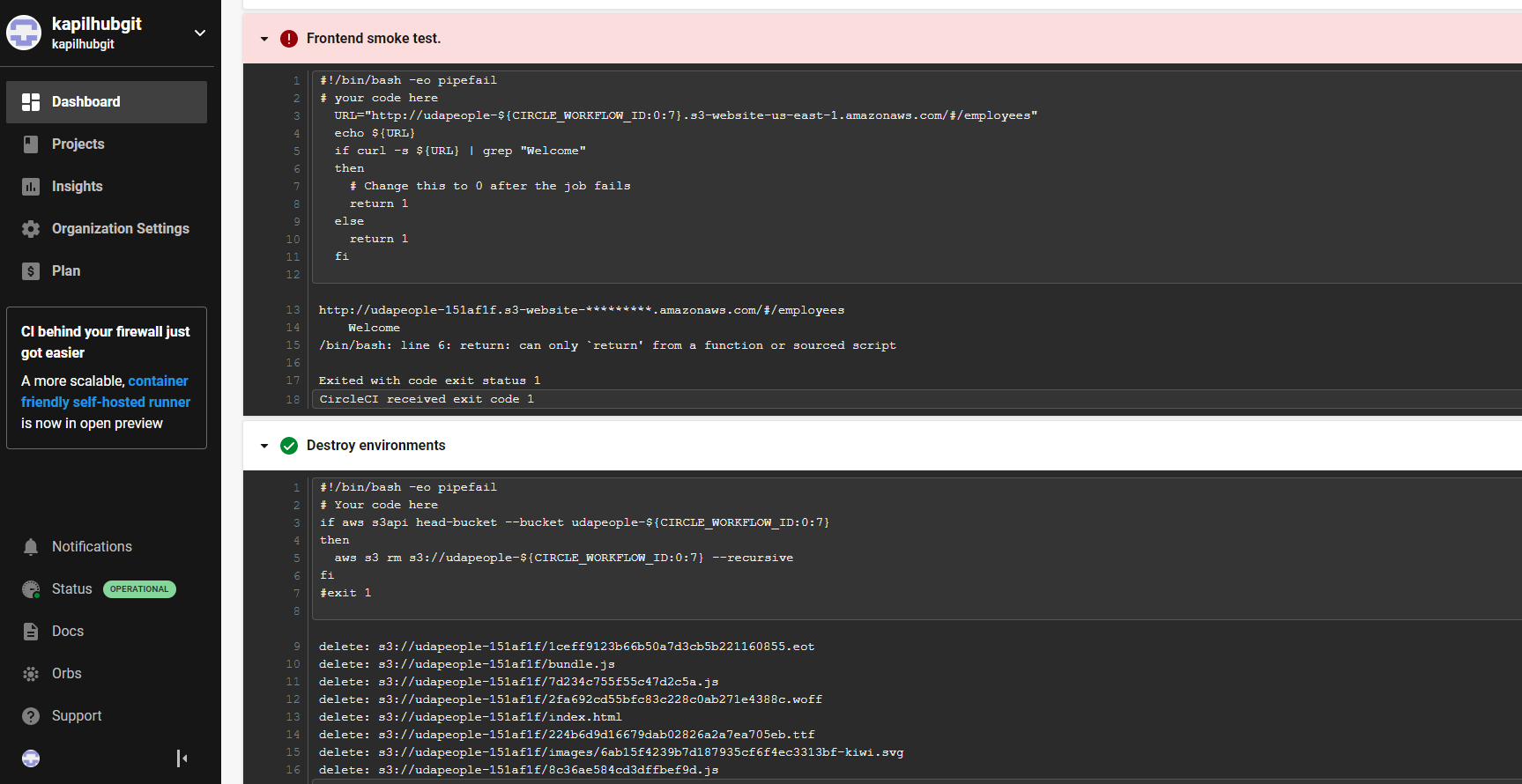
**Console output of appropriate failure for infrastructure creation job (using CloudFormation).**[SCREENSHOT05]

****

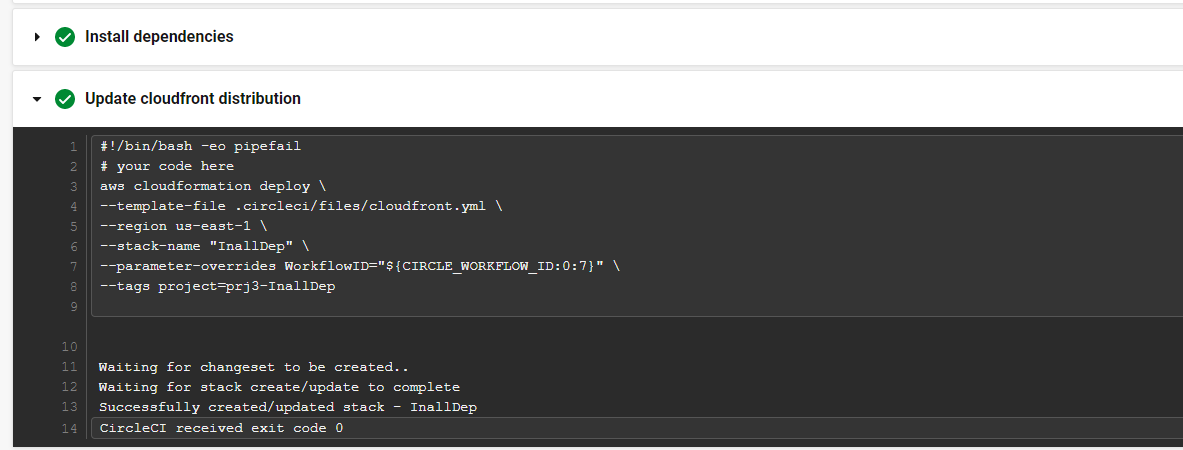
**Console output of a smoke test job that is failing appropriately.**[SCREENSHOT06]

****

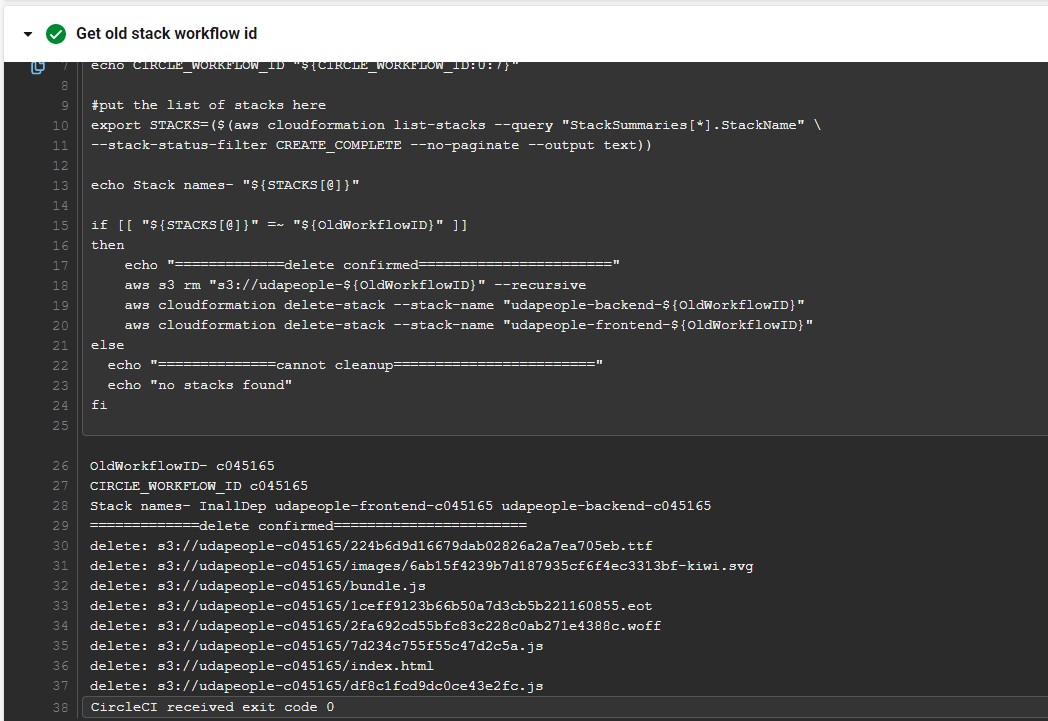
**Console output of a successful rollback after a failed smoke test.**[SCREENSHOT07]

****

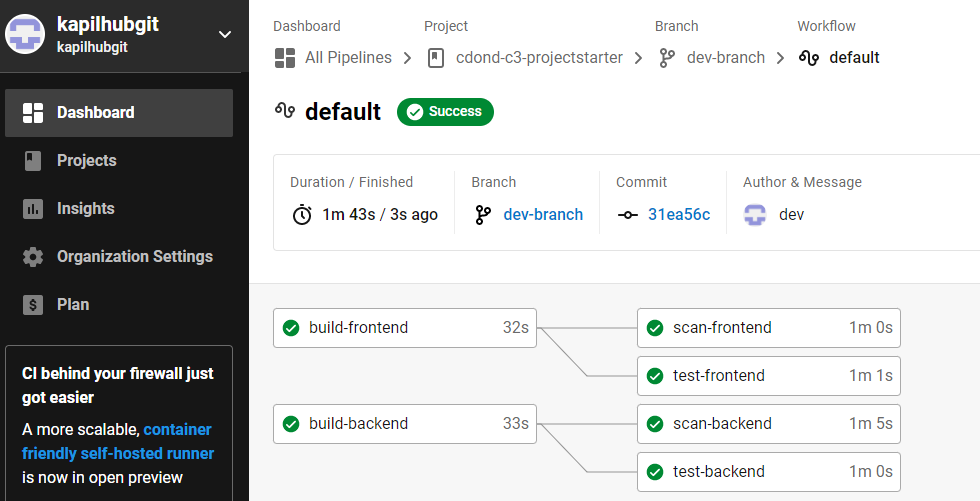
**Console output of successful promotion of new version to production in CloudFront.**[SCREENSHOT08]

****

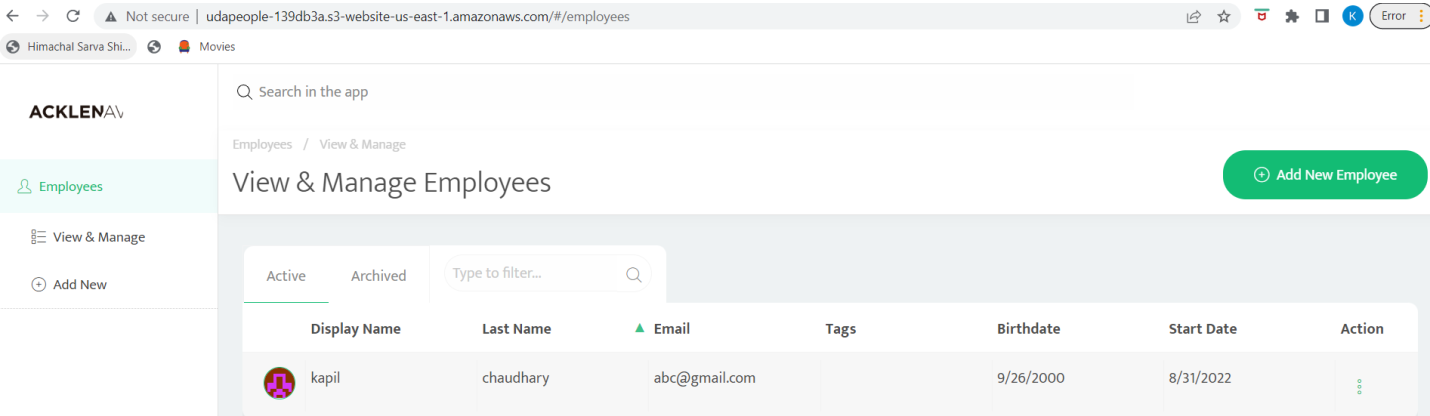
**Console output of successful cleanup job that removes old S3 bucket and EC2 instance.**[SCREENSHOT09]



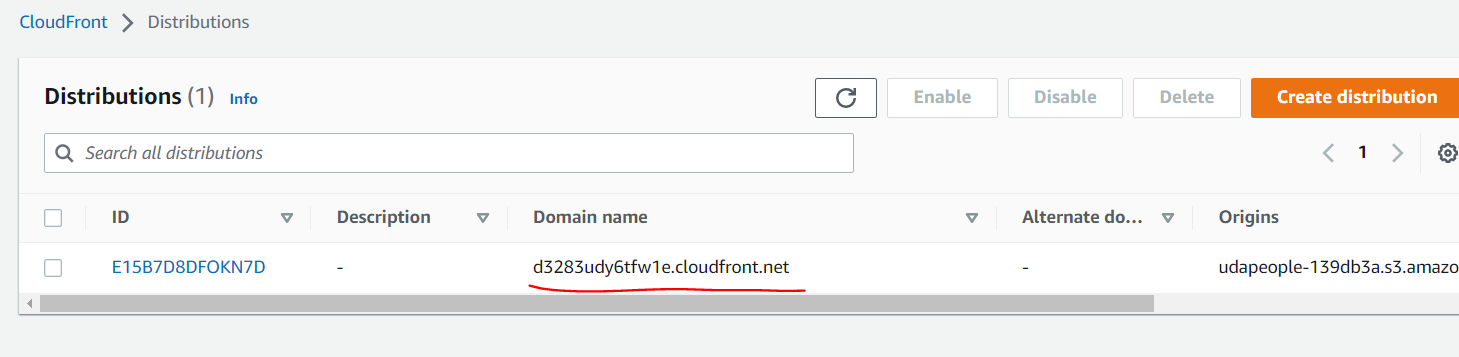
**Evidence that the deploy jobs only happen on the master branch.**[SCREENSHOT10]

****

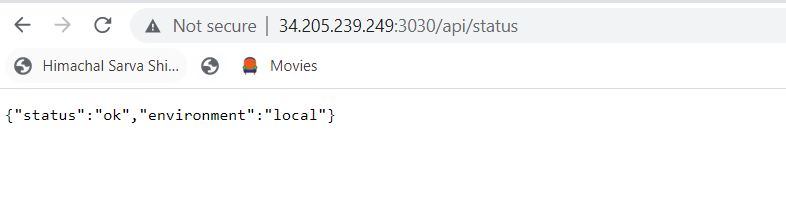
**Evidence of deployed and functioning front-end application in an S3 bucket**[URL02]**.**

****

**Evidence of deployed and functioning front-end application in CloudFront.**[URL03\_SCREENSHOT]

****

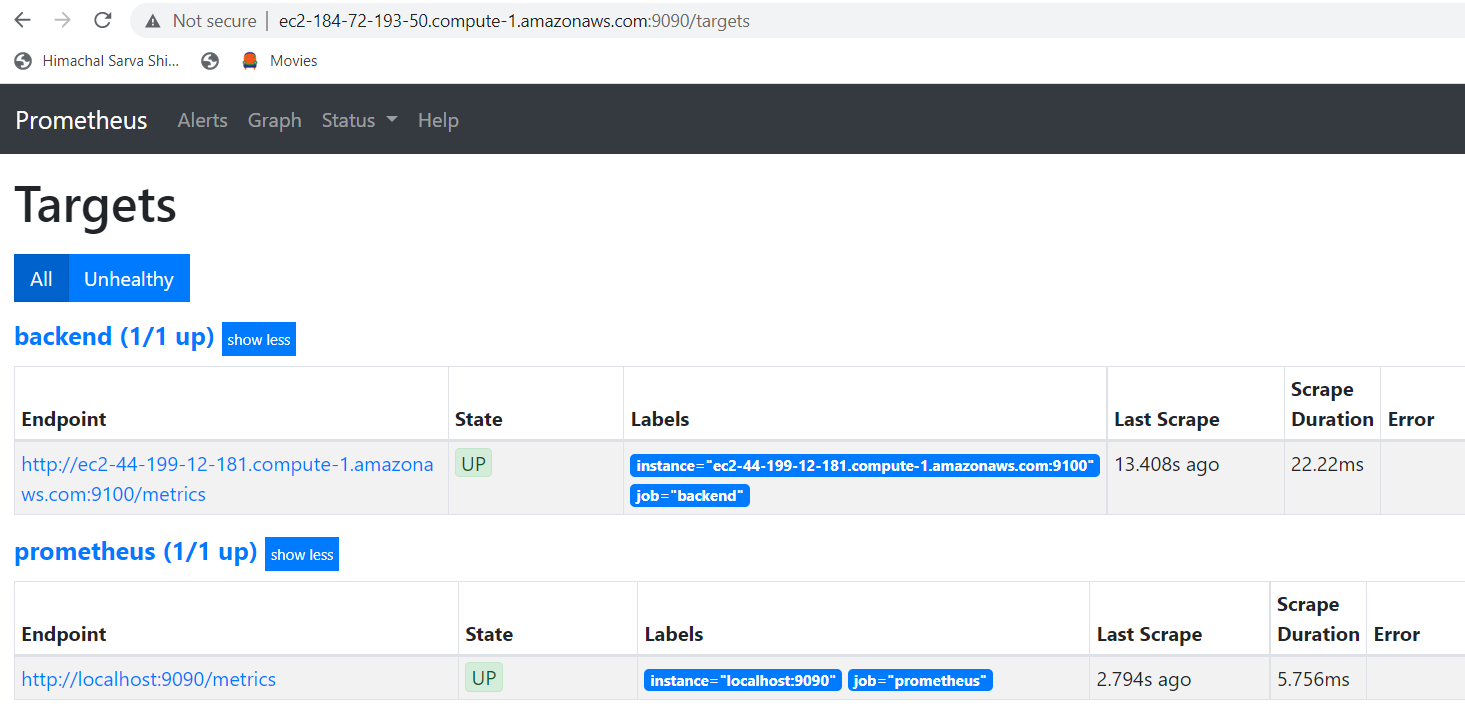
**Evidence of healthy back-end application.**[URL04\_SCREENSHOT]

****

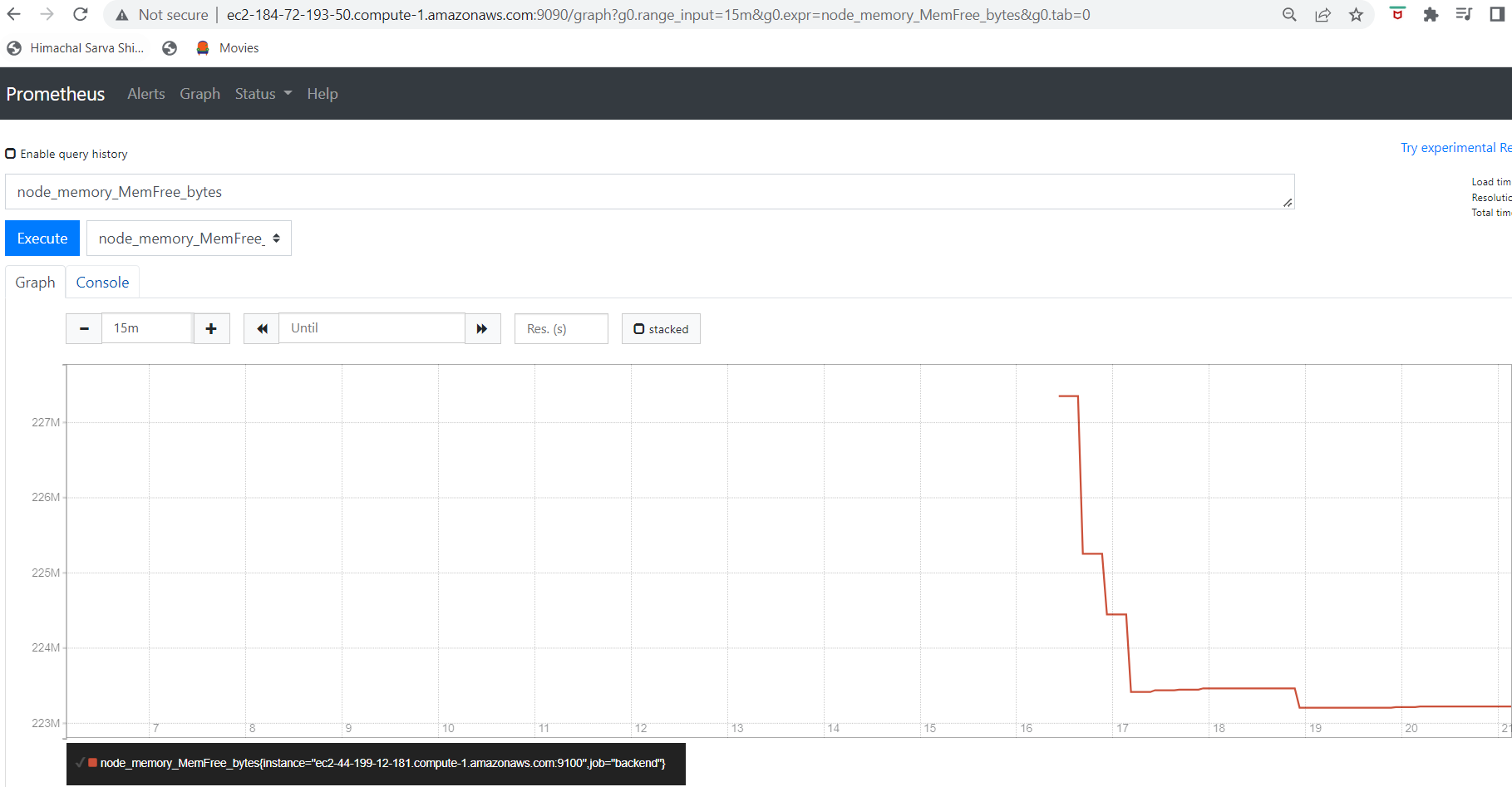
# Section 3: Turn Errors into Sirens

**CRITERIA:Surface critical server errors for diagnosis using centralized logging.**

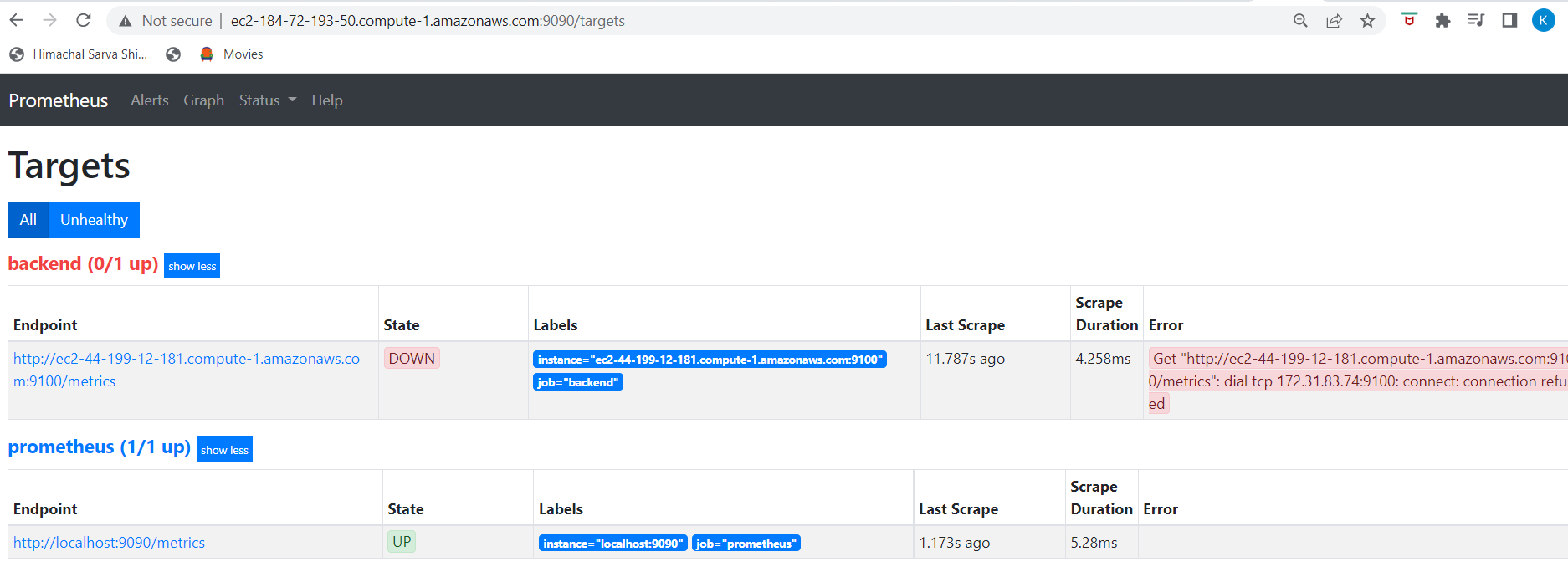
**Evidence of Prometheus Server.**[URL05\_SCREENSHOT]**.**

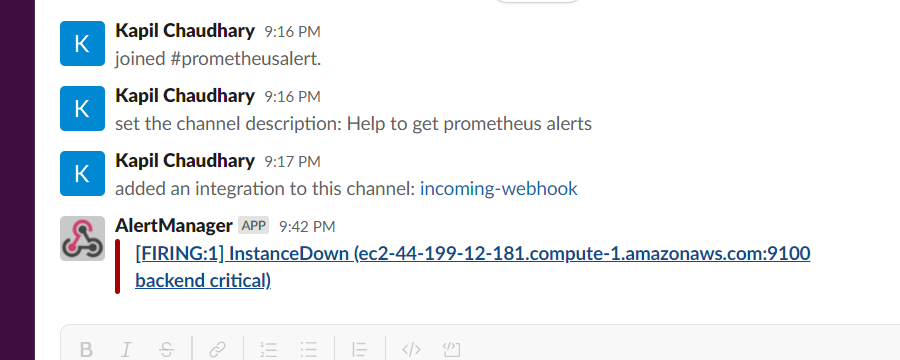
****

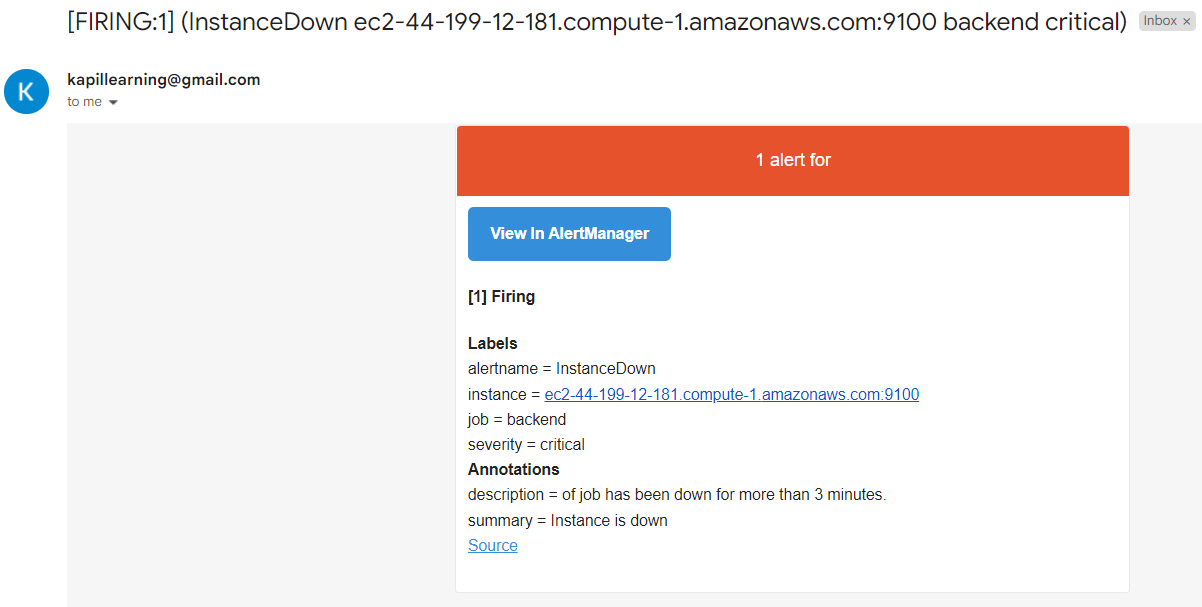
**Evidence that Prometheus is monitoring memory, cpu and disk usage of EC2 instances.**[SCREENSHOT11]

****

**Evidence that Prometheus and AlertManager send alerts when certain conditions exist in the EC2 instance.**[SCREENSHOT12]

****

****

****